Large Scale Debugging on Titan and Mira with Allinea DDT

David Lecomber
Allinea Software
david@allinea.com

Why?

Allinea’s Tools

allinea DDT

Reduce Development Time

allinea MAP

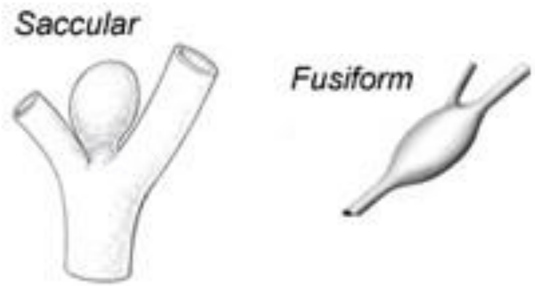
Increase Application Performance

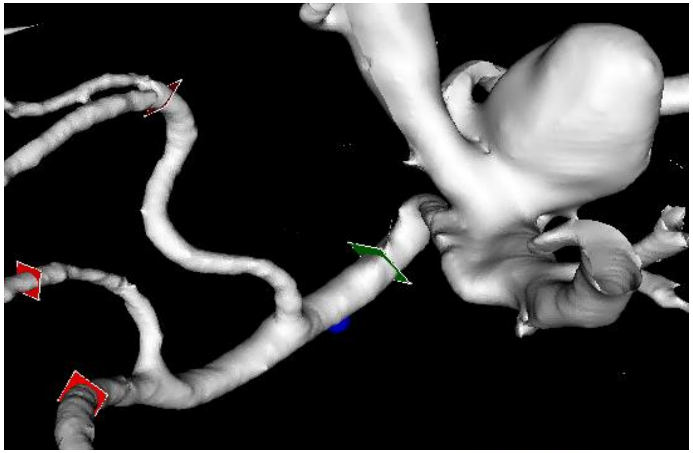
allinea Performance Reports

Understand Application Behaviour

HPC could be brain surgery

- Brain aneurysms
  - 2-5% of population – most are undiagnosed
  - 30,000 rupture in US each year – 40% fatal
  - Early discovery and treatment increases survival rates
- Neurosurgery as HPC
  - MRI provides the blood vessel structure
  - Intra-cranial blood flow and pressures is just complex CFD
  - Full brain 3D model is 2-10GB geometry





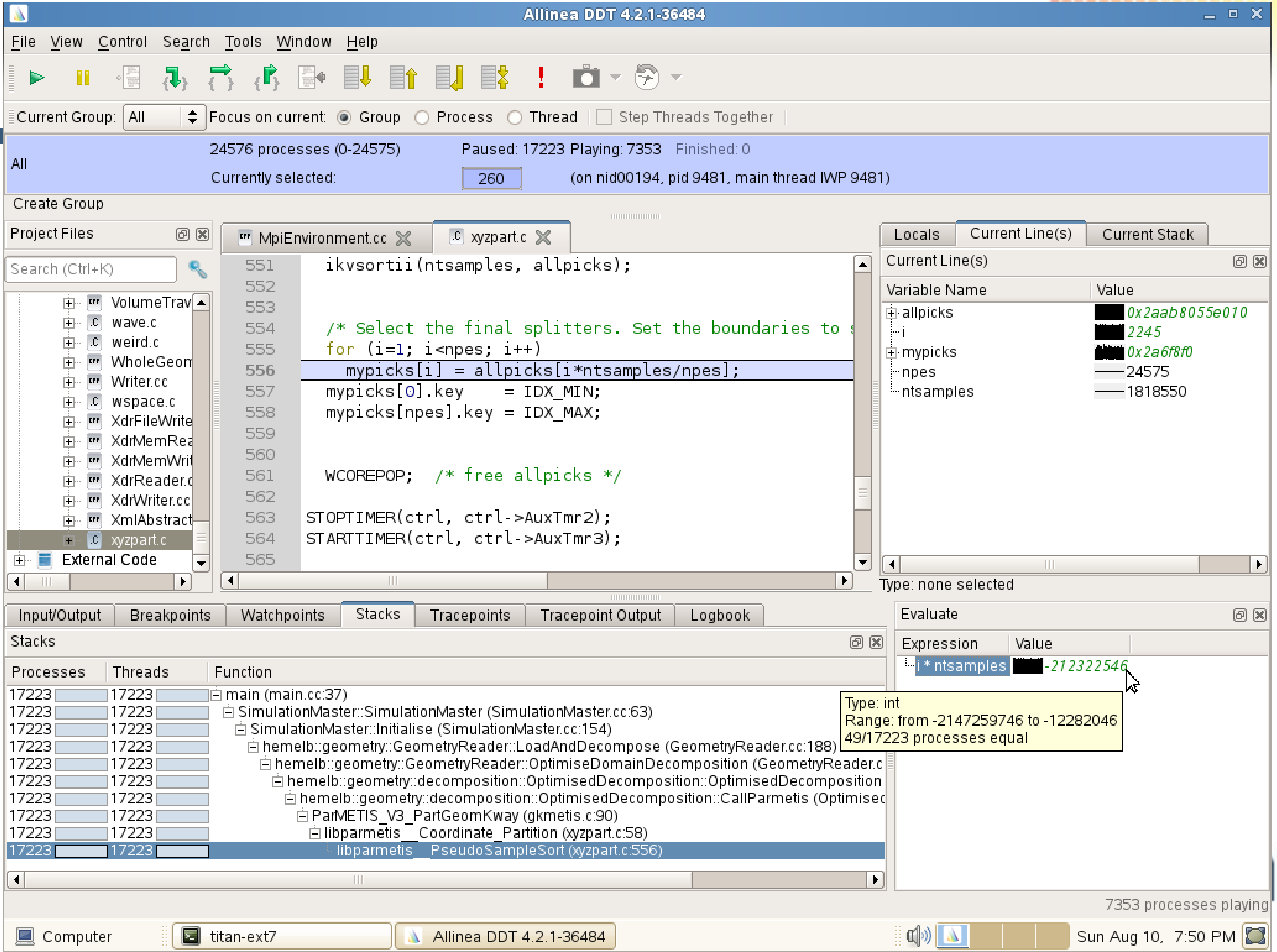
allinea
www.allinea.com

Impact of Petascale and Beyond

- Individualized HPC
  - – Patient’s MRI scan enables surgical decision: whether to operate, how to operate, …
  - – Circle of Willis requires super-Petascale ~~machine~~ software
  - – Need answer in minutes or hours
- Machines can do 20 PetaFLOPs
  - – Super-Petascale will be affordable soon
  - – Software has to scale

Real scaling challenge

- Crashes at 49,152 cores on Cray XC30
  - Error message “Terminated”. Thanks.
- Now what?
  - Try other (inferior?) partitioner?
  - Invest weeks in bug fix by trial and error?
  - Write own partitioning library?
- Why use a debugger?
  - It’s about time



Allinea DDT 4.2.1-36484

File View Control Search Tools Window Help

Current Group: All | Focus on current: ● Group ○ Process ○ Thread □ Step Threads Together

All | 24576 processes (0-24575) | Paused: 17223 Playing: 7353 | Finished: 0

Currently selected: 260 (on nid00194, pid 9481, main thread IWP 9481)

Create Group

Project Files

Search (Ctrl+K)

- VolumeTrav
- wave.c
- weird.c
- WholeGeon
- Writer.cc
- wspace.c
- XdrFileWrite
- XdrMemRea
- XdrMemWrit
- XdrReader.c
- XdrWriter.cc
- XmlAbstract
- xyzpart.c
- External Code

MpiEnvironment.cc | xyzpart.c

```
  ikvsortii(ntsamples, allpicks);


  /* Select the final splitters. Set the boundaries to s
  for (i=1; i<npes; i++)
    mypicks[i] = allpicks[i*ntsamples/npes];
  mypicks[0].key    = IDX_MIN;
  mypicks[npes].key = IDX_MAX;


  WCOREPOP;  /* free allpicks */

STOPTIMER(ctrl, ctrl->AuxTmr2);
STARTTIMER(ctrl, ctrl->AuxTmr3);
```

Locals | Current Line(s) | Current Stack

Current Line(s)

Variable Name	Value
allpicks	0x2aab8055e010
i	2245
mypicks	0x2a6f8f0
npes	24575
ntsamples	1818550

Type: none selected

Input/Output | Breakpoints | Watchpoints | Stacks | Tracepoints | Tracepoint Output | Logbook

Stacks

Processes	Threads	Function
17223	17223	main (main.cc:37)
17223	17223	SimulationMaster::SimulationMaster (SimulationMaster.cc:63)
17223	17223	SimulationMaster::Initialise (SimulationMaster.cc:154)
17223	17223	hemelb::geometry::GeometryReader::LoadAndDecompose (GeometryReader.cc:188)
17223	17223	hemelb::geometry::GeometryReader::OptimiseDomainDecomposition (GeometryReader.c
17223	17223	hemelb::geometry::decomposition::OptimisedDecomposition::OptimisedDecomposition
17223	17223	hemelb::geometry::decomposition::OptimisedDecomposition::CallParmetis (Optimised
17223	17223	ParMETIS_V3_PartGeomKway (gkmetis.c:90)
17223	17223	libparmetis__Coordinate_Partition (xyzpart.c:58)
17223	17223	libparmetis__PseudoSampleSort (xyzpart.c:556)

Evaluate

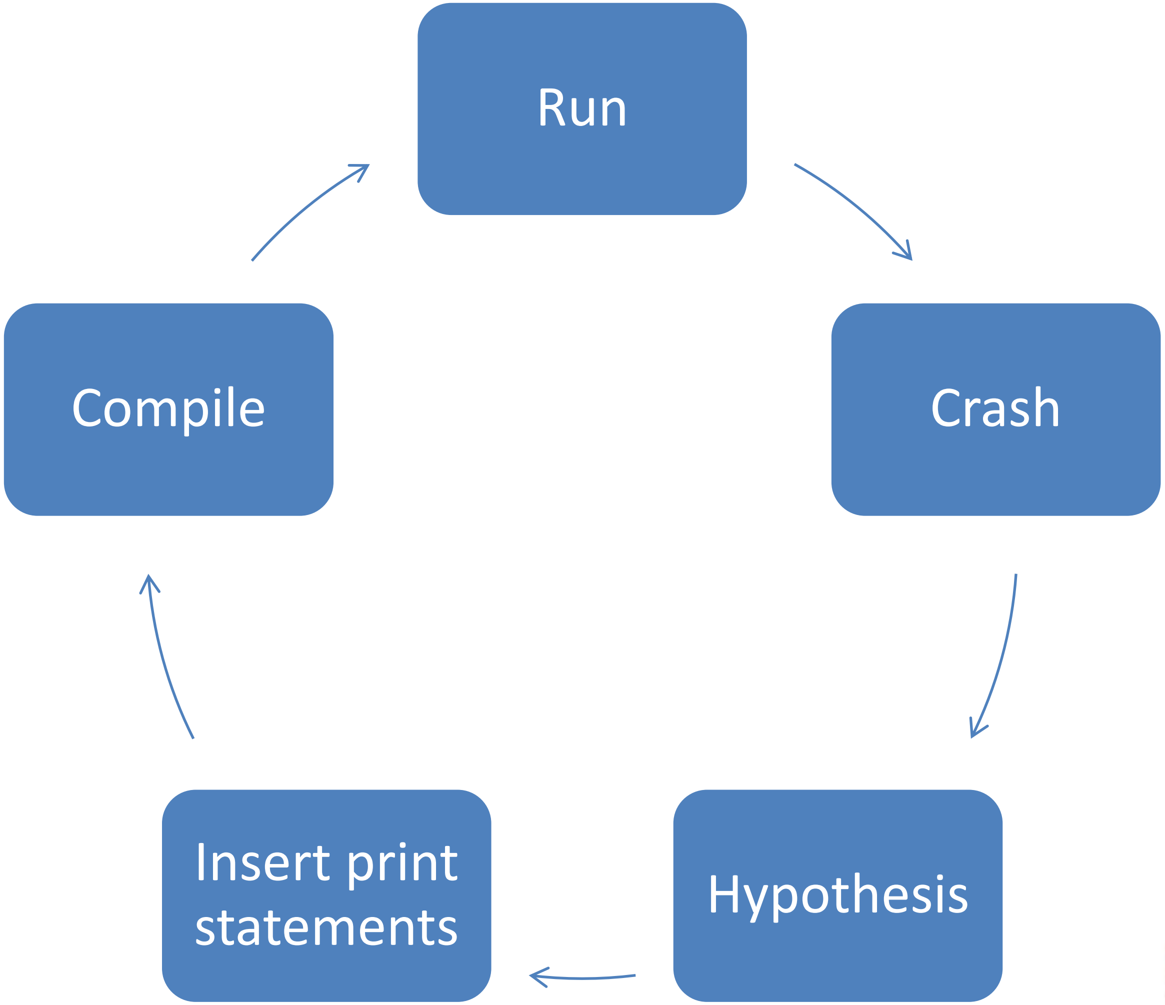
Expression	Value
i * ntsamples	-212322546

Type: int
Range: from -2147259746 to -12282046
49/17223 processes equal

7353 processes playing

Computer | titan-ext7 | Allinea DDT 4.2.1-36484 | Sun Aug 10, 7:50 PM

Debugging in practice…



Run

Crash

Hypothesis

Insert print statements

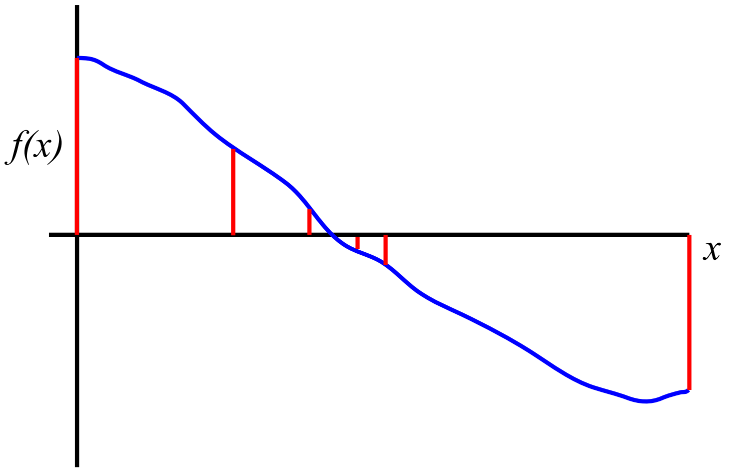
Compile

Some types of bug

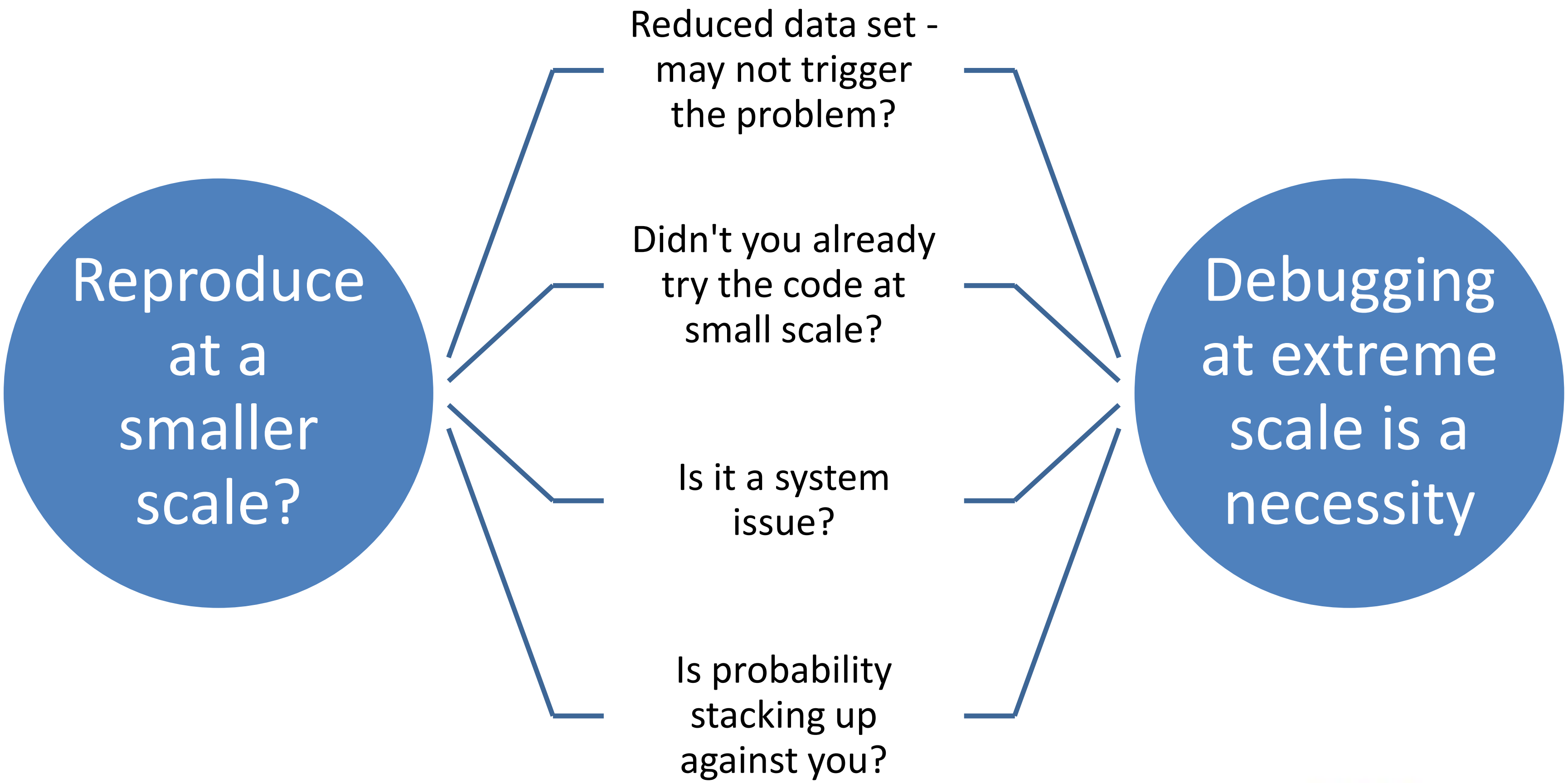
Bohrbug	Steady, dependable bug
Heisenbug	Vanishes when you try to debug (observe)
Mandelbug	Complexity and obscurity of the cause is so great that it appears chaotic
Schroedinbug	First occurs after someone reads the source file and deduces that it never worked, after which the program ceases to work

Print statement debugging

- The first debugger: print statements
  - Each process prints a message or value at defined locations
  - Diagnose the problem from evidence and intuition
- A long slow process
  - Analogous to bisection root finding
- Broken at modest scale
  - Too much output – too many log files



Bug fixing as scale increases



Reproduce at a smaller scale?

Reduced data set - may not trigger the problem?

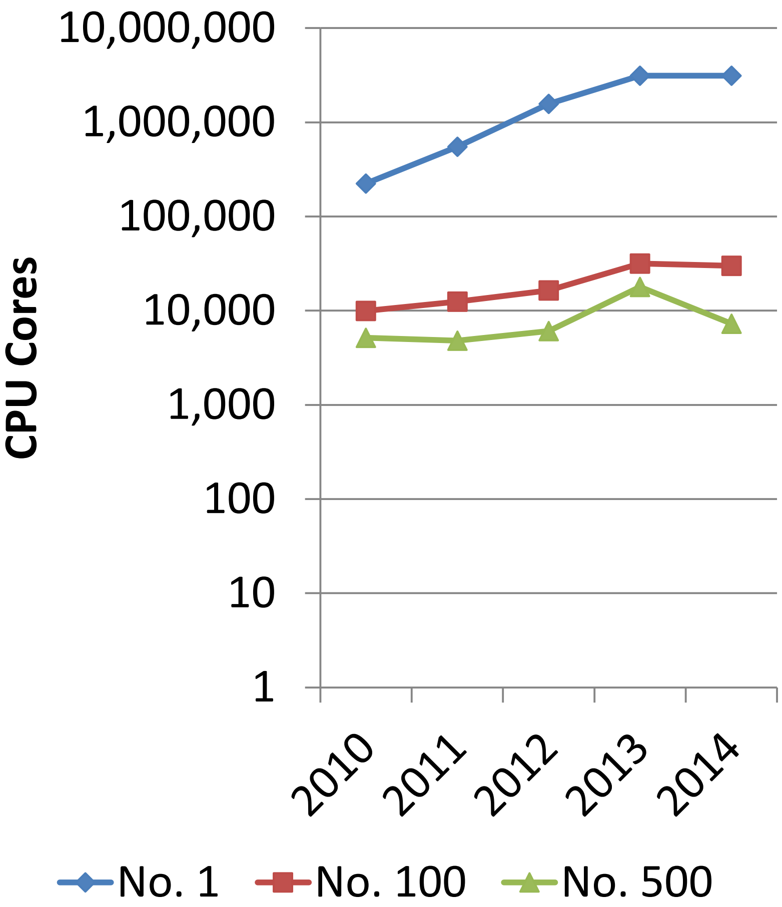
Didn't you already try the code at small scale?

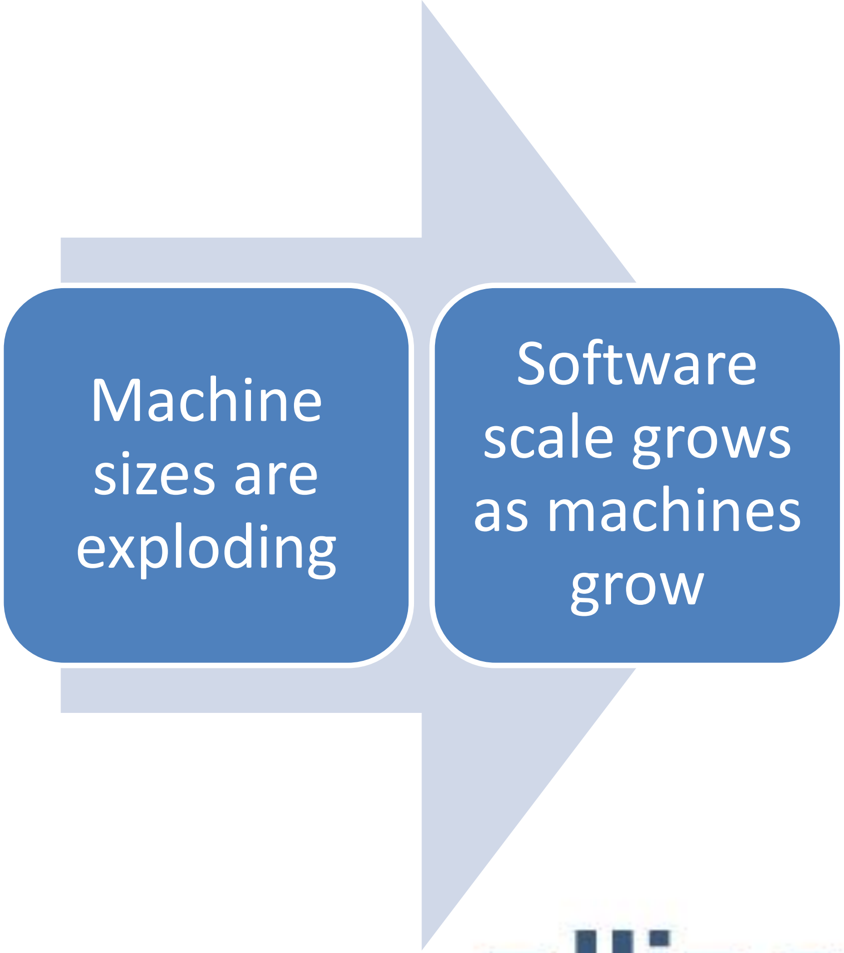
Is it a system issue?

Is probability stacking up against you?

Debugging at extreme scale is a necessity

Extreme machines are everywhere





Machine sizes are exploding

Software scale grows as machines grow

www.allinea.com

Titan and Mira

Titan

- 18,688 nodes
- 18,688 NVIDIA Kepler K20 GPUs
- 299,008 CPU cores
- 50,233,344 CUDA cores

Mira

- 49,152 nodes
- 786,432 cores
- 3,145,728 hardware threads

Does the printf workflow “work”?

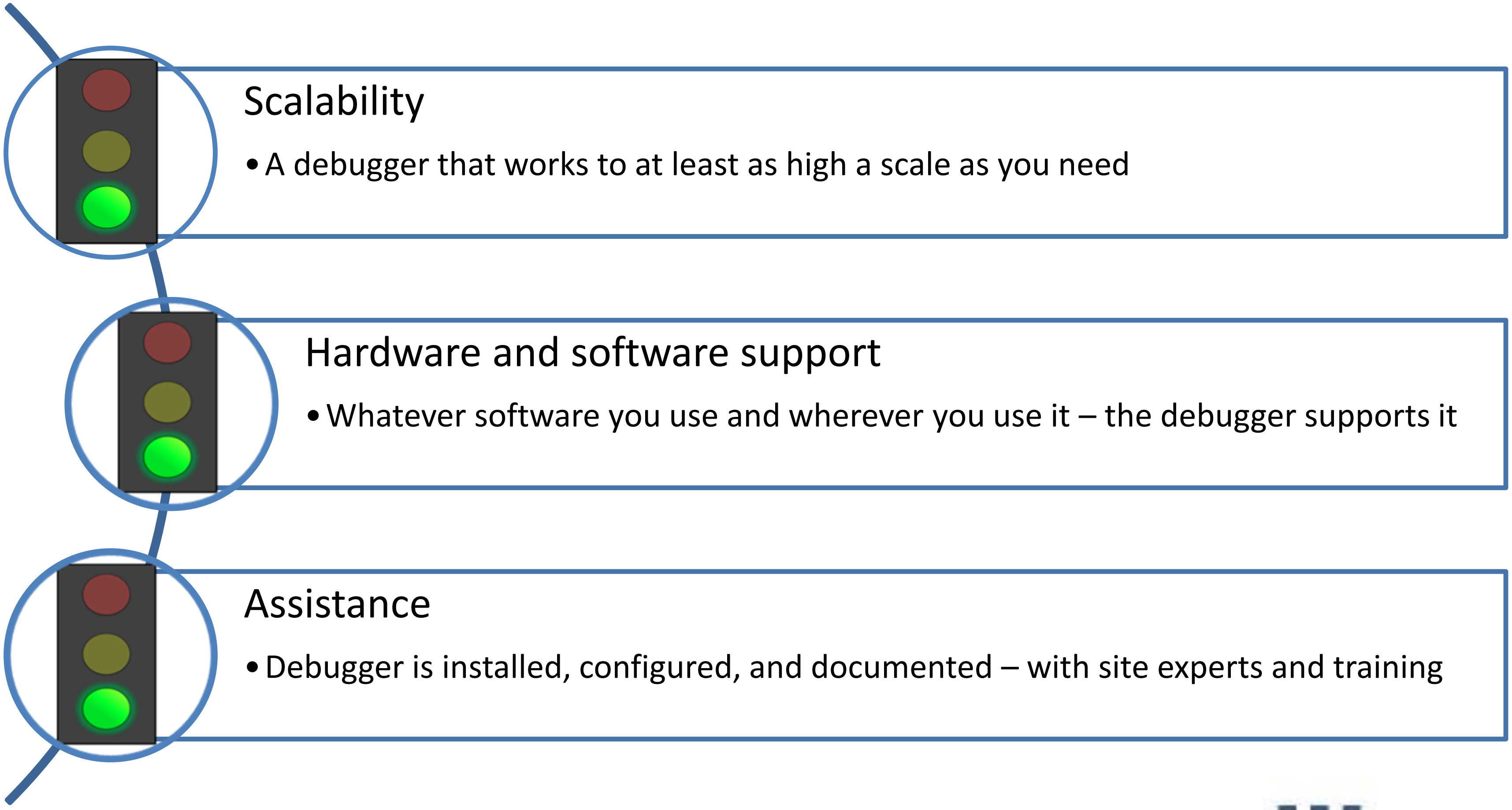
ALCF, OLCF and Allinea deliver

2009 - Allinea and Oak Ridge begin collaboration to provide super-Petascale debugging

2010 - Allinea and Argonne collaboration to extend scaling to BlueGene systems

2013 - Mira and Titan full size debugging in place

What you should expect (demand!) for debugging at scale



Scalability

- A debugger that works to at least as high a scale as you need

Hardware and software support

- Whatever software you use and wherever you use it – the debugger supports it

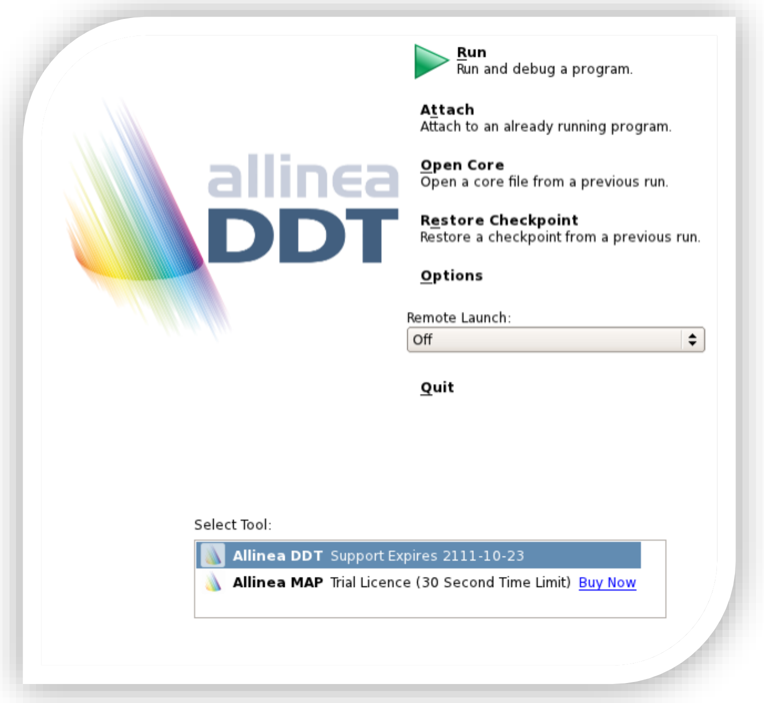
Assistance

- Debugger is installed, configured, and documented – with site experts and training

allinea
www.allinea.com

Allinea DDT
Fix software problems, fast

- **Powerful graphical debugger designed for :**
  - C/C++, Fortran, UPC, …
  - MPI, OpenMP and mixed-mode code
  - Accelerators and coprocessors: CUDA and Intel Xeon Phi



Run
Run and debug a program.

Attach
Attach to an already running program.

Open Core
Open a core file from a previous run.

Restore Checkpoint
Restore a checkpoint from a previous run.

Options

Remote Launch: Off

Quit

Select Tool:

Allinea DDT Support Expires 2111-10-23
Allinea MAP Trial Licence (30 Second Time Limit) Buy Now

- **Unified interface with Allinea MAP :**
  - One interface eliminates learning curve
  - Spend more time on your results

- **Slash your time to debug**
  - Reproduces and triggers your bugs instantly
  - Helps you easily understand where issues come from quickly
  - Helps you to fix them as swiftly as possible

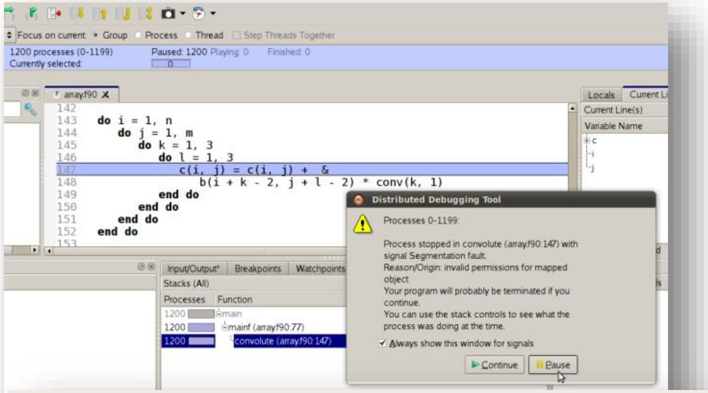
allinea
www.allinea.com

Allinea DDT: Scalable debugging by design

- ***Where* did it happen?**
  - Allinea DDT leaps to source automatically
  - Merges stacks from processes and threads
- ***How* did it happen?**
  - Some faults evident instantly from source
- ***Why* did it happen?**
  - Real-time data comparison and consolidation
  - Unique “Smart Highlighting” – colouring differences and changes
  - Sparklines comparing data across processes

– ***Force* crashes to happen?**

  - Memory debugging makes many random bugs appear every time



1200 processes (0-1199) Paused: 1200 Playing: 0 Finished: 0
Currently selected: 0

```
142
143   do i = 1, n
144      do j = 1, m
145         do k = 1, 3
146            do l = 1, 3
147               c(i, j) = c(i, j) +  &
148                  b(i + k - 2, j + l - 2) * conv(k, 1)
149            end do
150         end do
151      end do
152   end do
153
```

Stacks (All)

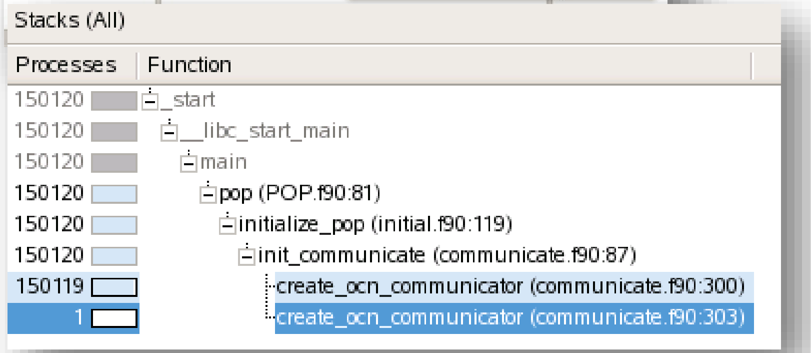
```
Processes  Function
1200       main
1200         mainf (array.f90:77)
1200           convolute (array.f90:147)
```

Distributed Debugging Tool

Processes 0-1199:

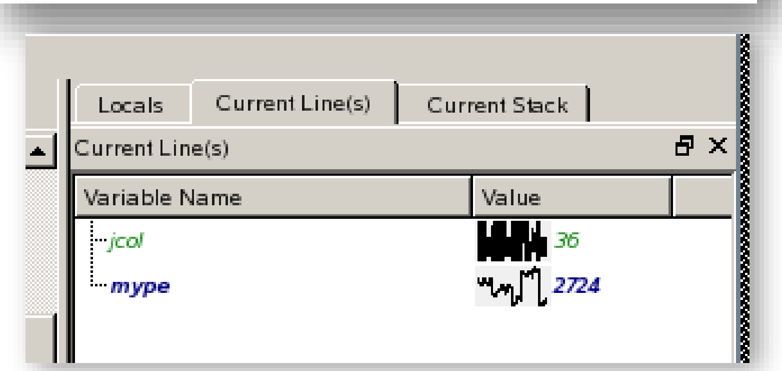
Process stopped in convolute (array.f90:147) with signal Segmentation fault.
Reason/Origin: invalid permissions for mapped object
Your program will probably be terminated if you continue.
You can use the stack controls to see what the process was doing at the time.

☑ Always show this window for signals



Stacks (All)

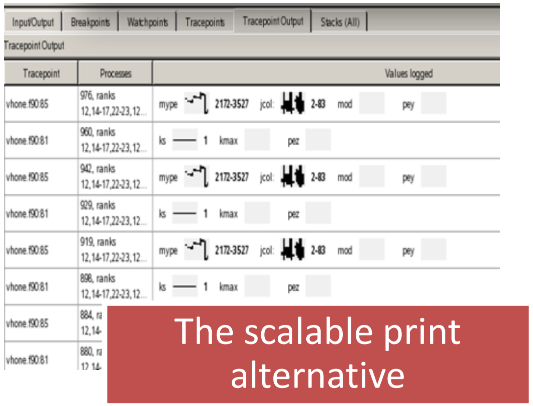
```
Processes  Function
150120     _start
150120       __libc_start_main
150120         main
150120           pop (POP.f90:81)
150120             initialize_pop (initial.f90:119)
150120               init_communicate (communicate.f90:87)
150119                 create_ocn_communicator (communicate.f90:300)
     1                 create_ocn_communicator (communicate.f90:303)
```



Current Line(s)

Variable Name	Value
jcol	36
mype	2724

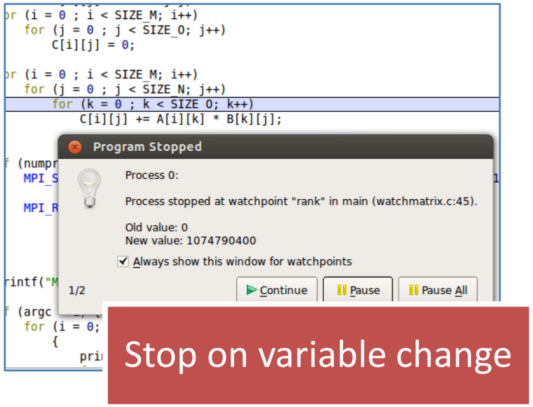
Five great things to try with Allinea DDT



Tracepoint Output

Tracepoint	Processes	Values logged
vhone.f90:85	976, ranks 12,14-17,22-23,12...	mype 2172-3527 jcol: 2-83 mod pey
vhone.f90:81	960, ranks 12,14-17,22-23,12...	ks 1 kmax pez
vhone.f90:85	942, ranks 12,14-17,22-23,12...	mype 2172-3527 jcol: 2-83 mod pey
vhone.f90:81	929, ranks 12,14-17,22-23,12...	ks 1 kmax pez
vhone.f90:85	919, ranks 12,14-17,22-23,12...	mype 2172-3527 jcol: 2-83 mod pey
vhone.f90:81	898, ranks 12,14-17,22-23,12...	ks 1 kmax pez
vhone.f90:85	884, ra 12,14-	
vhone.f90:81	880, ra 12,14-	

The scalable print alternative



```
or (i = 0 ; i < SIZE_M; i++)
   for (j = 0 ; j < SIZE_O; j++)
       C[i][j] = 0;

or (i = 0 ; i < SIZE_M; i++)
   for (j = 0 ; j < SIZE_N; j++)
       for (k = 0 ; k < SIZE_O; k++)
           C[i][j] += A[i][k] * B[k][j];


f (numpr
   MPI_S

   MPI_R



rintf("M

f (argc
   for (i = 0;
       {
           pri
```

Program Stopped

Process 0:

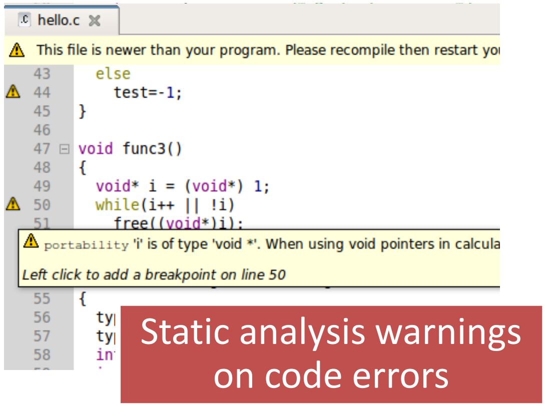
Process stopped at watchpoint "rank" in main (watchmatrix.c:45).

Old value: 0
New value: 1074790400

☑ Always show this window for watchpoints

1/2

Stop on variable change



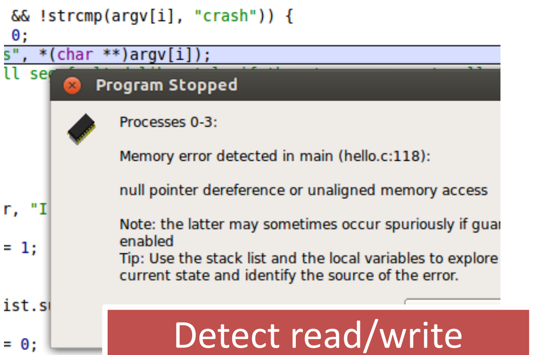
This file is newer than your program. Please recompile then restart yo

```
43     else
44       test=-1;
45   }
46
47   void func3()
48   {
49     void* i = (void*) 1;
50     while(i++ || !i)
51       free((void*)i);
55   {
56     ty
57     ty
58     in
```

portability 'i' is of type 'void *'. When using void pointers in calcula

Left click to add a breakpoint on line 50

Static analysis warnings on code errors



```
&& !strcmp(argv[i], "crash")) {
0;
s", *(char **)argv[i]);
ll se



r, "I

= 1;


ist.s

= 0;
```

Program Stopped

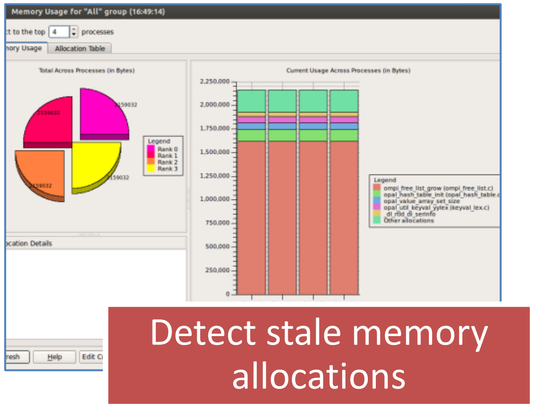
Processes 0-3:

Memory error detected in main (hello.c:118):

null pointer dereference or unaligned memory access

Note: the latter may sometimes occur spuriously if guar
enabled
Tip: Use the stack list and the local variables to explore
current state and identify the source of the error.

Detect read/write beyond array bounds



Memory Usage for "All" group (16:49:14)

t to the top 4 processes

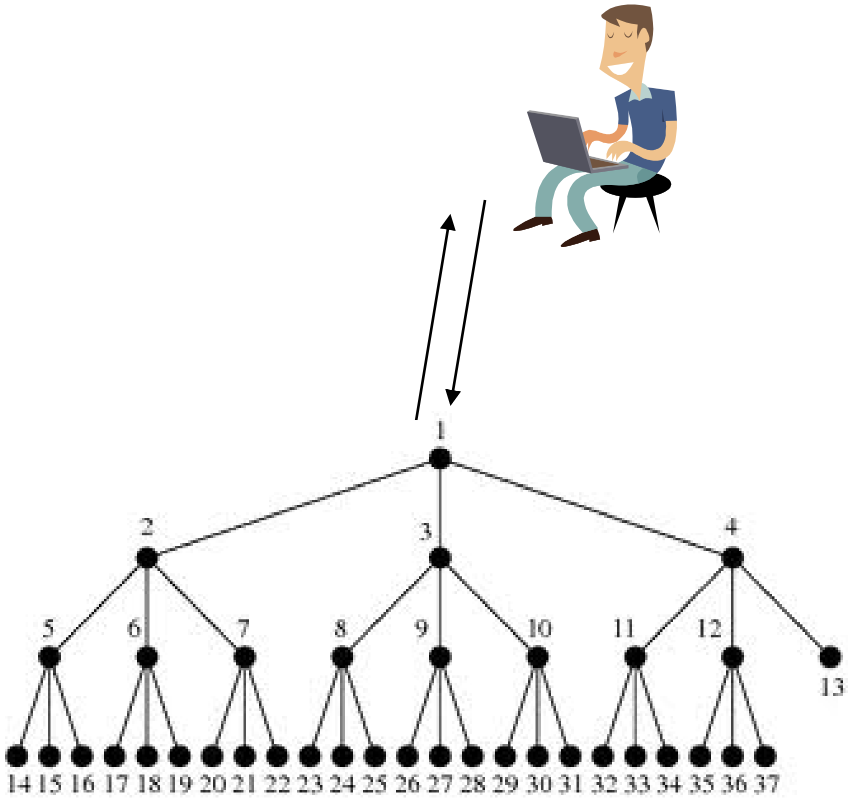
Detect stale memory allocations

www.allinea.com

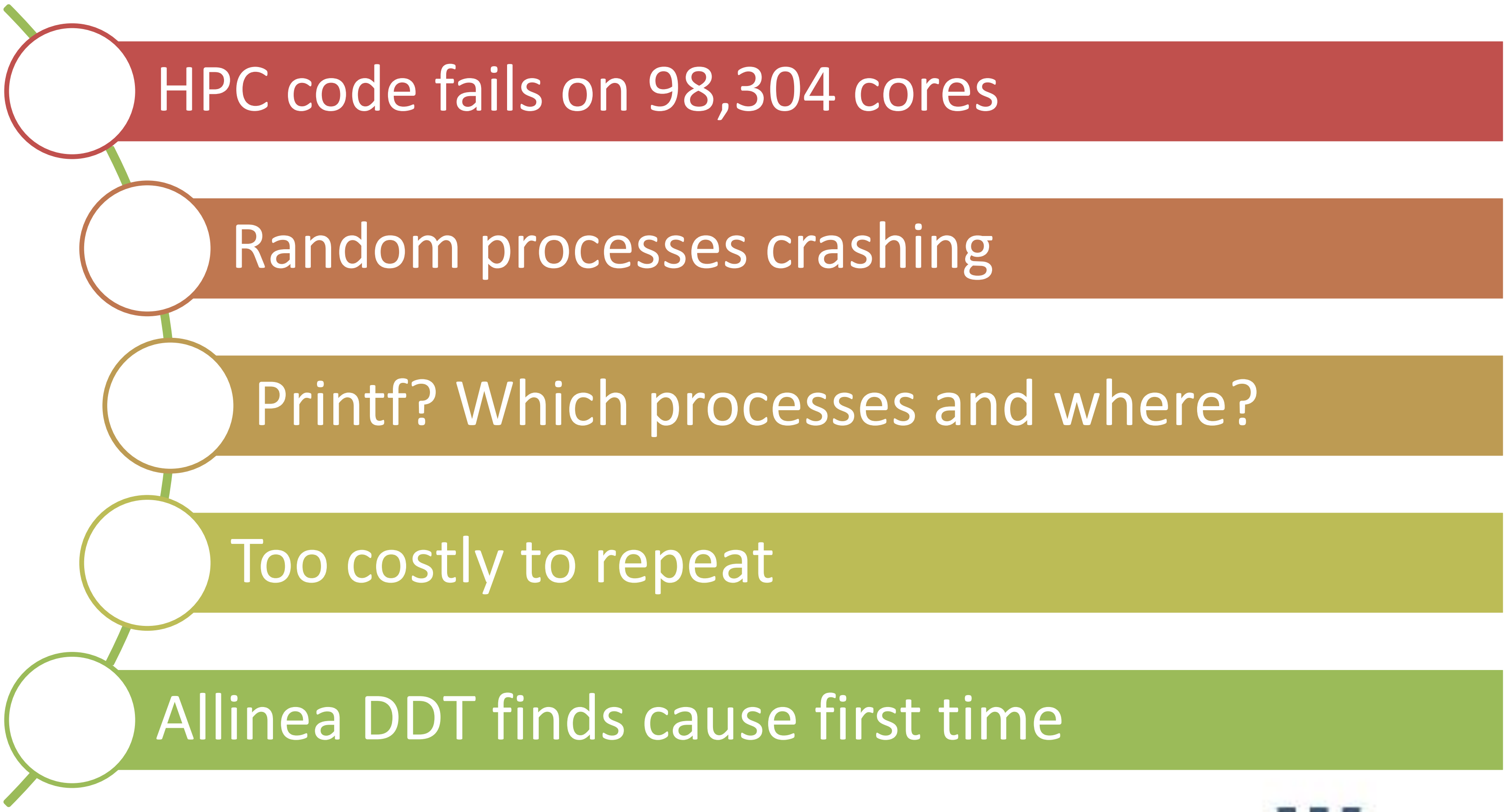
Beneath the Petascale Allinea DDT

- Scalable tree network
  - Sends bulk commands and merge responses
  - Aggregations maintain the essence of the information
  - Step 100,000 processes? 100-150ms

- Usability matters
  - The interface is as important as the speed
  - Focus on scalable components



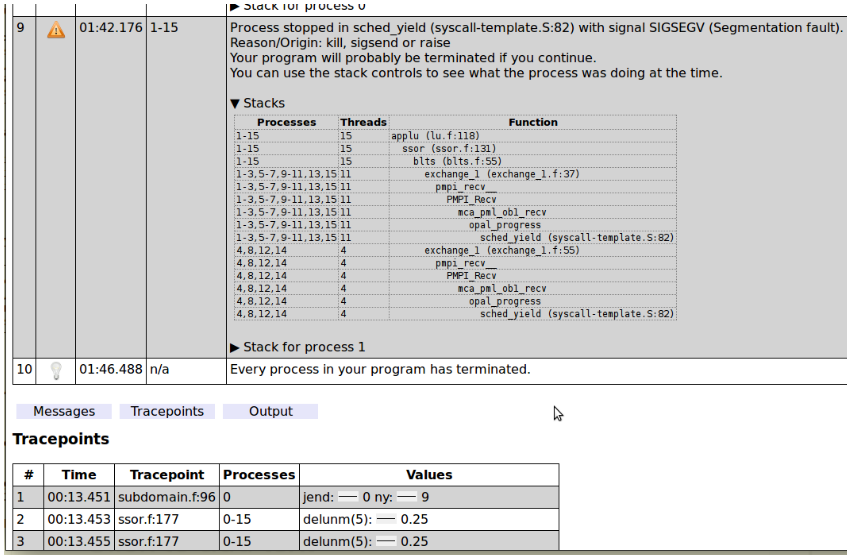
Example – ORNL’s Titan



- HPC code fails on 98,304 cores
- Random processes crashing
- Printf? Which processes and where?
- Too costly to repeat
- Allinea DDT finds cause first time

Offline debugging

- Interactive access difficult
- Use offline mode
  - Submit and forget
- Post-mortem analysis



▶ Stack for process 0

9 | ⚠ | 01:42.176 | 1-15 | Process stopped in sched_yield (syscall-template.S:82) with signal SIGSEGV (Segmentation fault).
Reason/Origin: kill, sigsend or raise
Your program will probably be terminated if you continue.
You can use the stack controls to see what the process was doing at the time.

▼ Stacks

```
Processes            Threads  Function
1-15                 15       applu (lu.f:118)
1-15                 15         ssor (ssor.f:131)
1-15                 15           blts (blts.f:55)
1-3,5-7,9-11,13,15   11             exchange_1 (exchange_1.f:37)
1-3,5-7,9-11,13,15   11               pmpi_recv__
1-3,5-7,9-11,13,15   11                 PMPI_Recv
1-3,5-7,9-11,13,15   11                   mca_pml_ob1_recv
1-3,5-7,9-11,13,15   11                     opal_progress
1-3,5-7,9-11,13,15   11                       sched_yield (syscall-template.S:82)
4,8,12,14            4              exchange_1 (exchange_1.f:55)
4,8,12,14            4                pmpi_recv__
4,8,12,14            4                  PMPI_Recv
4,8,12,14            4                    mca_pml_ob1_recv
4,8,12,14            4                      opal_progress
4,8,12,14            4                        sched_yield (syscall-template.S:82)
```

▶ Stack for process 1

10 | 01:46.488 | n/a | Every process in your program has terminated.

Tracepoints

#	Time	Tracepoint	Processes	Values
1	00:13.451	subdomain.f:96	0	jend: — 0 ny: — 9
2	00:13.453	ssor.f:177	0-15	delunm(5): — 0.25
3	00:13.455	ssor.f:177	0-15	delunm(5): — 0.25

Example – ANL Mira

- HPC code fails on 16,384 cores
- Code abandoned – bug couldn’t be fixed
- Machine too busy for interactive debugging
- Allinea DDT offline mode runs bug case overnight
- Found error in initialization

Interlude: Local Demonstration

- Simple persistent hanging
  - – Stepping through a code

- Process count dependent hanging:
  - – Attaching to the running job

Getting started on Titan

- How?

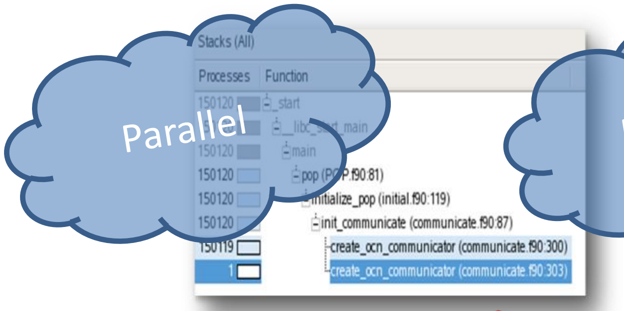
```
module load ddt
ddt
```

- Congratulations, you are now ready to debug.

Getting started on Mira/Tukey

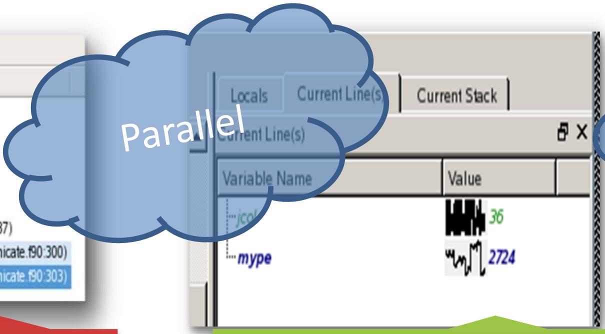
- Install local client on your laptop
  - www.allinea.com/downloads
    - Linux – installs full set of tools
    - Windows, Mac – just a remote client to the remote system
  - Run the installation and software
  - “Connect to remote host”
  - Hostname:
    - username@cetus.alcf.anl.gov
    - username@tukey.alcf.anl.gov
  - Remote installation directory: /soft/debuggers/ddt
  - Click Test
- Congratulations you are now ready to debug.

Favorite Allinea DDT Features for Scale



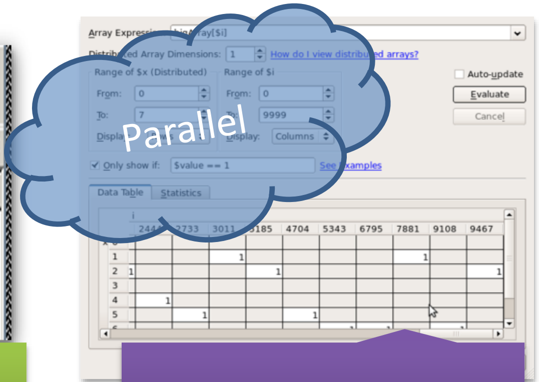
Processes	Function
150120	_start
150120	__libc_start_main
150120	main
150120	pop (POP.f90:81)
150120	initialize_pop (initial.f90:119)
150120	init_communicate (communicate.f90:87)
150119	create_ocn_communicator (communicate.f90:300)
1	create_ocn_communicator (communicate.f90:303)

Parallel stack view



Variable Name	Value
icol	36
mype	2724

Automated data comparison: sparklines



Array Expression: bigArray[$i]

Distributed Array Dimensions: 1

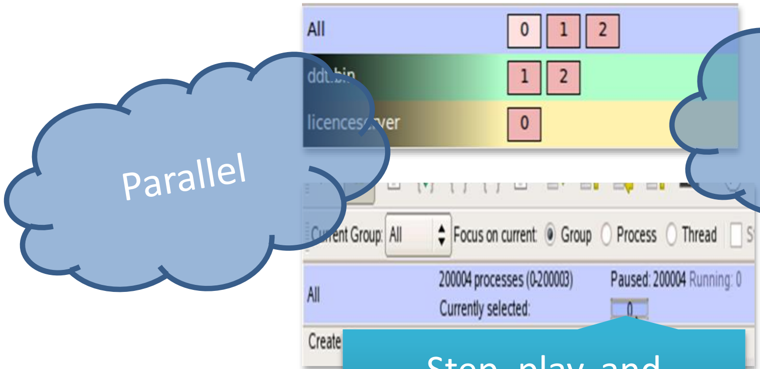
Range of $x (Distributed) From: 0 To: 7 Display: Rows

Range of $i From: 0 To: 9999 Display: Columns

☑ Only show if: $value == 1

x	2444	2733	3011	3185	4704	5343	6795	7881	9108	9467
1			1					1		
2				1						1
3										
4	1									
5		1			1					

Parallel array searching

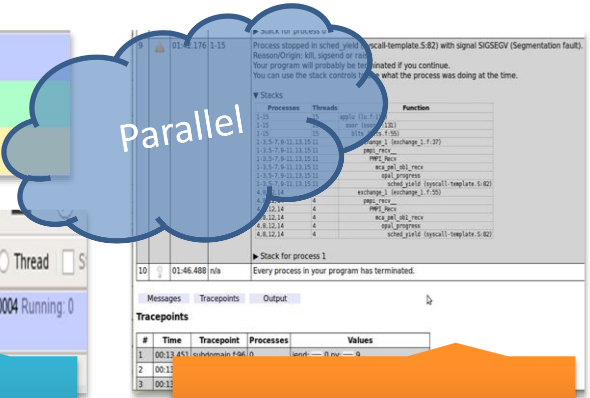


Current Group: All Focus on current: ● Group ○ Process ○ Thread

All 200004 processes (0-200003) Paused: 200004 Running: 0

Currently selected: 0

Step, play, and breakpoints



9 01:42.176 1-15 Process stopped in sched_yield (syscall-template.S:82) with signal SIGSEGV (Segmentation fault).
Reason/Origin: kill, sigsend or raise
Your program will probably be terminated if you continue.
You can use the stack controls to see what the process was doing at the time.

▼ Stacks

Processes	Threads	Function
1-15	15	applu (lu.f:1?)
1-15	15	ssor (ssor.f:131)
1-15	15	blts (blts.f:55)
1-3,5-7,9-11,13,15	11	exchange_1 (exchange_1.f:37)
1-3,5-7,9-11,13,15	11	pmpi_recv__
1-3,5-7,9-11,13,15	11	PMPI_Recv
1-3,5-7,9-11,13,15	11	mca_pml_ob1_recv
1-3,5-7,9-11,13,15	11	opal_progress
1-3,5-7,9-11,13,15	11	sched_yield (syscall-template.S:82)
4,8,12,14	4	exchange_1 (exchange_1.f:55)
4,8,12,14	4	pmpi_recv__
4,8,12,14	4	PMPI_Recv
4,8,12,14	4	mca_pml_ob1_recv
4,8,12,14	4	opal_progress
4,8,12,14	4	sched_yield (syscall-template.S:82)

▶ Stack for process 1

10 01:46.488 n/a Every process in your program has terminated.

Tracepoints

#	Time	Tracepoint	Processes	Values
1	00:13.451	subdomain.f:96	0	iend: — 0 nx: — 9
2	00:13			
3	00:13			

Offline debugging

www.allinea.com

Summary

Debugging at scale is not difficult

- 300,000 cores is as easy as 30 cores
- The user interface is vital to success

Debugging at scale is not slow

- High performance debugging – at Mira and Titan scale
- Logarithmic performance

Stable, in production and well supported

- Routinely used over 100,000 cores